

IP TELEPHONY BASED DANGER ALERT COMMUNICATION SYSTEM AND ITS IMPLEMENTATION

Filip REZAC¹, Filip KOLENOVSKY², Martin MIKULEC¹, Miroslav VOZNAK¹, Jan ROZHON¹,
Jiri VYCHODIL¹, Karel TOMALA¹, Jakub SAFARIK¹

¹Department of Telecommunications, Faculty of Electrical Engineering and Computer Science,
VSB–Technical University of Ostrava, 17. listopadu 15, 708 00 Ostrava-Poruba, Czech Republic

²Department of Computer Science, Faculty of Electrical Engineering and Computer Science,
VSB–Technical University of Ostrava, 17. listopadu 15, 708 00 Ostrava-Poruba, Czech Republic

filip.rezac@vsb.cz, filip.kolenovsky@vsb.cz, martin.mikulec@vsb.cz, miroslav.voznak@vsb.cz, jan.rozhon@vsb.cz,
jiri.vychodil@vsb.cz, karel.tomala@vsb.cz, jakub.safarik@vsb.cz

Abstract. *The paper deals with development of a web application allowing to deliver pre-recorded voice messages by using SIP generator. The developed application is a part of complex system, which has been evolved in Dpt. of Telecommunications, Technical University of Ostrava for last three years. Our intent is focused on disaster management, the message, which should be delivered within specified time span, is typed in the application and text-to-speech module ensures its transformation to a speech format, after that a particular scenario or warned area is selected and a target group is automatically unloaded. For this purpose, we have defined XML format for delivery of phone numbers which are located in the target area and these numbers are obtained from mobile BTS's (Base transmission stations). The advantage of this approach lies in controlled delivery and finally the list of unanswered calls is exported and these users are informed via SMS. In the paper, the core of the danger alert system is described including algorithms of the voice message delivery.*

Keywords

Asterisk, Danger Alert System, PHP, SIP, Sipp, VoIP.

1. Introduction

Our system is based on using SIP call generator to generate and distribute voice messages directly to the end device (IPphone, cellphone, fixed line, etc.), [1]. The benefit of such communication compared to others is the fact that it uses a phone call and therefore, it is possible to get feedback who accepted the message and to improve efficiency of alert system. The whole system will be based in the data centre of a telecommunications operator

and will be accessible to the crisis centre's staff. A staff member logs into the system created by us, loads the pre-recorded alert and other parameters and sends the request. The output of the application are SIP messages which are sent into communication server, it can be based on open-source softswitch, such as Asterisk or OpenSIPS [2], [3].

The softswitch processes the messages and starts initiating calls to the end users according to the pre-defined parameters. The end user obtains a better understanding and sufficient information to solve the situation. If the end user does not answer the call (missed call, phone switched off), the system arranges to re-send the message and re-initiate the call with the end user concerned.

2. Technology

Primarily, the system is designed to alert population to a dangerous situation. Naturally, it can be used in other areas of activities. The system has been de-signed as LAMP server [4], [5], [6]. The crisis centre' staff can operate it through a web interface.

We suppose cooperation with mobile operators, they are able to deliver the list of numbers located in the target area. The list, which the staff, members enter is saved in the XML format and contains three columns - telephone number of the end user, BTS station where the subscriber is registered and a throughput rate of the BTS station [7]. The number of rows in the list equals the number of end users. The messages are entered in the system in the .wav or .pcap format [8]. This voice message is then sent out to all end users and played once the phone is answered. We have developed the conversion of a WAV container into a PCAP format using the Web application. The file is fed in using the input file field. Once the file is fed in by a user, the programme

reads the file's header. Based on the header, the codec used; number of channels, mono/stereo record, etc. are determined. The application supports up to two channels which can be converted into a mono record, μ -LAW, A-LAW and PCM codecs, 8 and 16 byte sample. The web server contains a simple form to input a file, to set up output codec and source and target IP address. Once these data have been filled in, the programme first reads the header and establishes whether the input codec is supported, the number of channels, etc. If a certain parameter does not correspond, the application returns an error message and the file is not converted. If everything is order, the file is converted and immediately offered for download by users.

The conversion source code, and the implementation of web pages are written in Nette Framework using PHP and jQuery. This allows for easy portability between systems and operation systems' independence and wide-spectrum of utilization. Our solution uses an Apache open source server. Data about samples stored are saved in a MySQL database.

Users can define for a file to be saved in a database [5]. To achieve that, the user needs to fill in the Information field. The file is saved under the name indicated in this field and is ready for further processing. The system contains a list of files saved together with information about the files. The files can still be reconverted after they have been saved.

The staff members may set the duration of the ringing. The call is not regarded as executed unless the end user answers the call. The remaining parameters entered into the system will be described further along with other algorithms which used these parameters. The system uses the open-source generator Sipp with pre-set call schemes in the XML format.

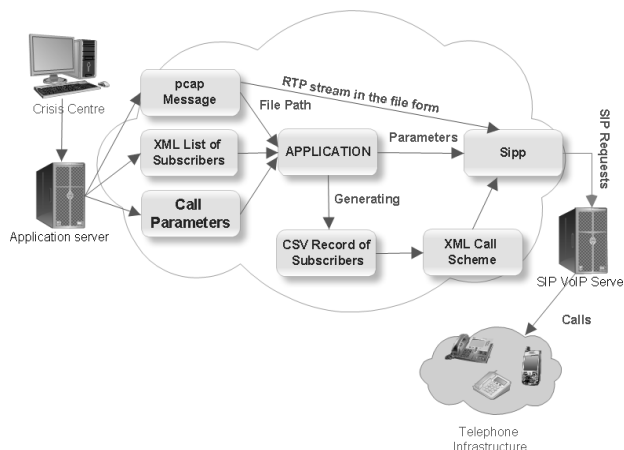


Fig. 1: Alert system scheme.

We met with Sipp in a research project in CESNET association and the acquired experience proved to be invaluable for this research [9], [10]. To be able to process parameters entered into Sipp and XML dynamically, we applied two methods. Using the first one, a correct parameter is assigned to the values entered

into the forms by the crisis centre staff, and it is then sent to Sipp. In order to dynamically switch the telephone numbers based on the loaded list of end users, values in the XML scheme for Sipp need to change dynamically too. This is the purpose of the .csv file, which is generated by the application directly after the list of end users has been loaded. This is where the XML scheme gets the telephone numbers and Sipp creates the SIP message for the softswitch. Figure 1 illustrates the scheme. Another issue addressed while developing the system was how many messages and subsequent calls are we able to generate at a particular moment.

During testing [11] it was found that the Sipp is currently the biggest limiting factor of the system. This open source tool is able within a single processor core correctly generate and maintain up to 200 SIP requests so that incoming answers were correctly processed. For higher values will begin to appear in INVITE requests flaws in their structure or length. The solution is an adaptation of Sipp tool to support parallel processes to optimize multi-core processors. With this change, which was involved by Liptel team, it is possible to reach up several thousands of calls simultaneously, depending on the number of processor cores [12]. In light of these facts, we can define the maximum number of generated calls. Voice distribution system automatically detects the number of processor cores and calculates what is the maximum value of generated SIP requests.

However, distributed sending of a larger amount of requests between participants without the linearly increasing the number of processor cores for the Sipp generator is possible only when the requests are divided into groups. The interval between the groups is set to 60 seconds. This period is determined by the maximum length of voice messages – 30s and a crisis centre staff can set the ringing for a maximum of 15 seconds. We get the sum equals to 45s. This means that the length of the call will never be longer than those 45s. From this, we can deduce that the distribution system is able to make 200 calls every 60 seconds using one CPU core. In the case of quad-core processor, this value is about 800 calls every 60 seconds. The following chapter will discuss algorithms that the system uses to work with the groups, missed calls and to calculate the total time required for distribution of messages.

3. Method

In the previous, chapter it was stated that the maximum number of calls that the system can generate and maintain, is 200 using one CPU core. This value is in the text and diagrams below referred to as C_{max} and may vary based on the number of CPU cores. For further progress, however, assume the usage of one CPU core. Once the staff member logs in s/he can enter the XML list of end users into the system. The system immediately calculates the total number of calls to be

generated $Creq$, number of groups to which the calls will be divided Gn and the estimated time to send all calls $Tsnd$ [s]:

$$Gn \cong \frac{Creq}{Cmax} . \quad (1)$$

The number of groups Gn is determined using formula (1), estimated time $Tsnd$ using formula (2):

$$Tsnd \cong 60 \cdot \left(\frac{Creq}{Cmax} \right). \quad (2)$$

If, for instance, staff member enters a list with 6800 end users, we will obtain the following values: $Creq = 6800$, $Gn = 34$ and $Tsnd = 2040$ s.

In case the threat of danger or natural disaster becomes real, staff has the exact time models setting the maximum time limits in which all end users should be alerted. This time value is inserted into the system form and is described as the maximum time to send all planned calls $Tmax$ [s]. Figure 2 illustrates two situations which can arise after the value is inserted. In the first case where $Tsnd \leq Tmax$, there is time remaining to re-send unanswered calls $Trem$ [s] after all calls have been sent for the first time. In this case, the system generates the value of time remaining $Trem$ and notifies the staff that during this time, the system is to automatically re-initiate calls which have not been answered in the first wave. Where the end user does not answer the call for the second time either, the calls are being re-initiated until $Trem = 0$. In the latter case, where $Tsnd > Tmax$, the system returns information that the time required to initiate all calls $Tsnd$ is longer that the maximum time for sending requests set by staff $Tmax$ and therefore, it is not possible to guarantee that all end users will be contacted and requests for unanswered calls re-generated. If staff want to contact all end users, or even set aside time to re-initiate unanswered calls, they need to increase $Tmax$ so that $Tmax > Tsnd$. This is how the staff get an idea of the time plan to generate calls and can thus make further steps to address the situation.

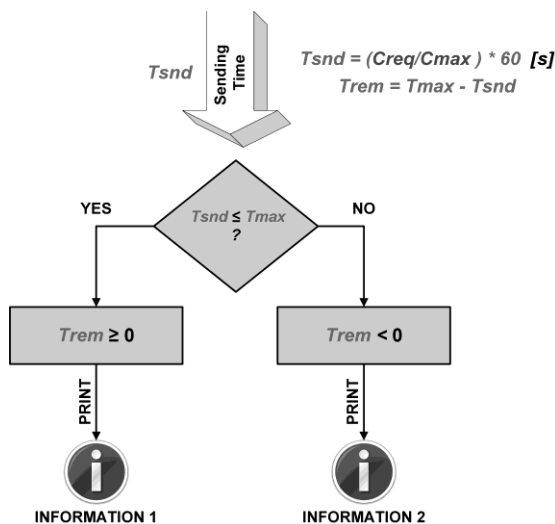


Fig. 2: Time diagram.

Other values which staff need to enter into the ready-made forms before calls can actually start to be generated are the duration of the ringing at the end user's side, login name and password to the SIP account and of course the location of the pre-recorded voice message.

Once all forms have been filled, the system can start initiating calls according to pre-set parameters. Staff members launch the call generation by pressing the SEND button. At first, the system logs into the IP switchboard at the SIP account created for this purpose and then it starts to generate SIP requests using the Sipp application.

Generating is carried out across individual groups ($G1-Gn$) with 60 seconds interval. Each group contains $Cmax$ call requests, the last group contains the remaining requests ($Cend$). By adding together all $Cmax$ and $Cend$, we obtain a total number of call requests $Creq$ (3). Requests are dispatched to the switchboard in sequence and the switchboard starts to initiate individual calls.

At this stage, the end device (cellphone, fixed line, etc.) of the end user starts to ring and keeps ringing for the time defined by staff in the system form. If the end user answers the call within this time span, a voice message containing information about the danger and how to address it is played to him/her. This call is marked as answered ($Cansw$). If the end user fails to answer the call during the defined period of ringing, or the end device is not available, the call is marked as unanswered ($Cmiss$).

Once all requests across all groups have been generated, the system adds together all unanswered calls (4). If $Trem > 0$, the system starts to generate SIP requests for call previously unanswered. Where $Cmiss > Cmax$, such calls are again subdivided into individual groups ($Gmiss1 - Gmissn$). The number of these groups can be determined using formula (5). Unanswered calls are re-initiated until $Trem = 0$ or until there are no unanswered calls $Cmiss = 0$:

$$Creq = Cmax_{G1} + Cmax_{G2} + \dots + Cmax_{Gn}, \quad (3)$$

$$Gmiss \cong \frac{Cmiss}{Cmax}, \quad (4)$$

$$Cmiss = \left\{ \begin{array}{l} Cmax_{Gmiss1} + Cmax_{Gmiss1} + \dots + Cend_{Gmissn} \\ Cmiss_{G1} + Cmiss_{G2} + \dots + Cmiss_{Gn} \\ Creq - Cansw \end{array} \right\}. \quad (5)$$

While generating, the system indicates the total number of requests $Creq$, number of groups Gn and $Gmiss$, total number of answered calls $Cansw$, number of unanswered calls $Cmiss$ and times $Tsnd$ and $Trem$. If an unexpected situation occurs, the whole process can be aborted by pressing the ABORT button. Figure 3 shows above mentioned procedures in the well-arranged diagram.

4. Features

The whole system (Fig. 4) was designed to distribute pre-recorded voice messages to alert people to danger or natural disaster. Its aim is to provide sufficient information to a large population on how to address a particular situation in the shortest time possible in a form which cannot be missed or disregarded. Depending on the content of the pre-recorded message, the system may not only function as a part of the early warning scheme, but it can also function as the basic infrastructure for other areas of human activities. Below we provide an overview of situations in which the system could be used:

- Alert system for population safety during natural disasters (floods, fires, earthquakes, wind storms, snow breaks, dangerous substance leakages), conflicts (attacks, wars, raids, army drills).
- Information system: during traffic congestion, to distribute poll results, to distribute election results, to announce sports competition results, to broadcast news (e.g. for visually impaired).
- Advertising system: to distribute advertisements, to address voters, to announce lottery winners.

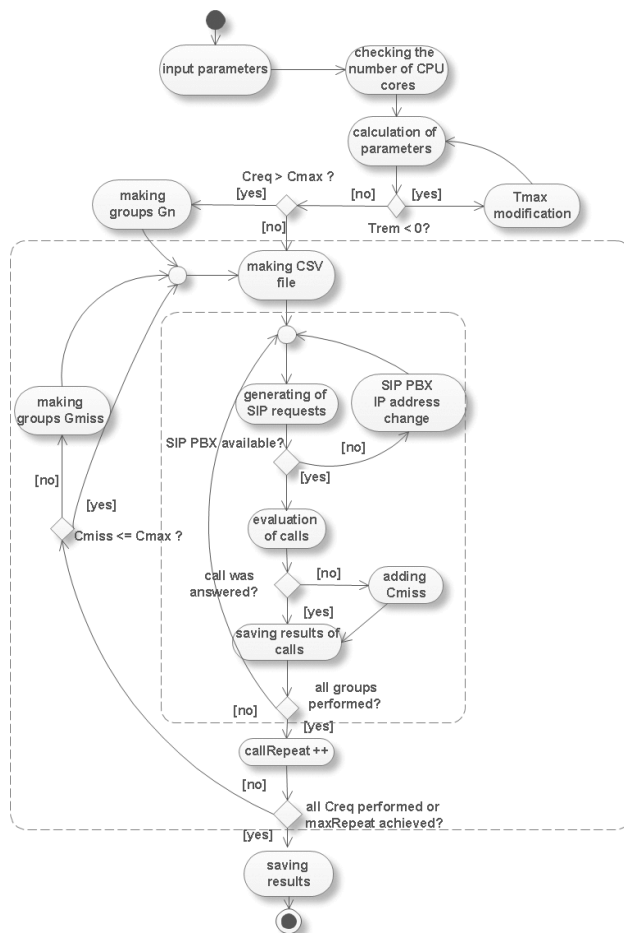


Fig. 3: Activity diagram.

Whatever situation the system is applied for, we can always reliably calculate the time necessary to

address the target group of end users. We use formula (2) where $Creq$ is the total number of end users to be targeted and $Cmax$ is set to 200. If it was necessary to inform all students at our university (24 058) about a certain event within 2 hours, our system would manage it. The same applies in case a natural disaster should occur – the system is able to advise community population within a certain time span depending on the population size.

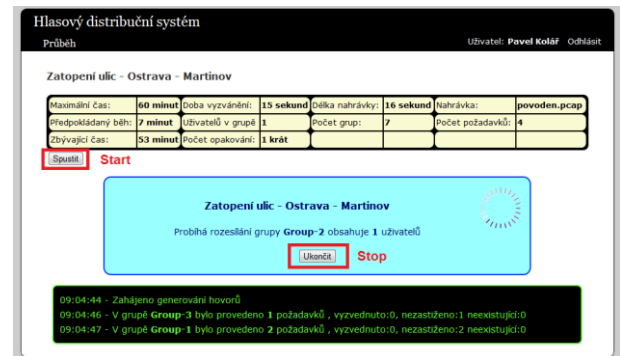


Fig. 4: Voice distribution system.

5. Conclusion

We developed a system which can distribute voice messages in the form of telephone calls using VoIP technology and SIP protocol fast and efficiently. It uses open-source application Sipp and any compatible softswitch enabling for registration of an SIP account. The benefit of using voice message to transfer information as opposed to other forms of information systems is that the information can be easily perceived by target audience, thus reducing the probability of it being missed. The system is controlled through a web interface, which makes it accessible from any machine with an Internet browser. At present, system is subject to intensive beta testing. We plan to implement several enhancements in the future, such as VoiceXML schemes for creating an interactive media dialog or a more user-friendly installation tool available through the web interface.

Acknowledgements

The research leading to these results has received funding from the European Community's Seventh Framework Programme INDECT (FP7/2007-2013) under the grant agreement no. 218086.

References

- [1] VOZNAK, M., F. REZAC and J. ZDRALEK. Danger Alert Communication System. In: *The 17th International Conference IWSSIP 2010*. Rio de Janeiro: Federal Fluminense University, 2010, pp. 231-234. ISBN 978-85-228-0565-5.

- [2] MEGGELEN, J., J. SMITH and L. MADSEN. *Asterisk: The Future of Telephony*. 2nd ed. Sebastopol: O'Reilly Media, 2007. ISBN 978-05-965-1048-0.
- [3] ONCALVES, Flavio E. *Building telephony systems with OpenSIPS 1.6: build scalable and robust telephony systems using SIP*. Birmingham: Packt Publishing, 2010. ISBN 978-1849510745.
- [4] LAURIE, Ben a Peter LAURIE. *Apache: the definitive guide*. 3rd ed. Sebastopol: O'Reilly, 2003. ISBN 0596002033.
- [5] DUBOIS, Paul. *MySQL*. 4th ed. Upper Saddle River: Addison-Wesley, 2009. ISBN 978-0672329388.
- [6] POWERS, David. *PHP Solutions: Dynamic Web Design Made Easy*. 2nd ed. Fully revised and updated content. New York: Friends of ED, 2010. ISBN 978-1430232490.
- [7] BESTAK, Robert. Evolution of mobile networks. In: *15th International Conference on Systems, Signals and Image Processing*. Bratislava: IEEE, 2008, pp. 25-28. ISBN 978-802272856-0. DOI: 10.1109/IWSSIP.2008.4604359.
- [8] VYCHODIL, J., M. VOZNAK, K. TOMALA, J. ROZHON and F. REZAC. Converting audio WAV container to PCAP format. In: *Conference proceedings Research in Telecommunication Technologies 2010*. Velke Losiny: VSB-Technical University of Ostrava, 2010, pp. 165-167. ISBN 978-80-248-2261-7.
- [9] REZAC, F., M. VOZNAK and J. RUZICKA. Security Risk in IP Telephony. In: *Proceedings CESNET Conference 2008*. Prague: CESNET, 2008, pp. 31-38. ISBN 978-80-904173-0-4.
- [10] VOZNAK, M. and F. REZAC. Implementation of SPAM over Internet Telephony and a Defence against this Attack. In: *32nd International Conference TSP*. Dunakiliti: Asszisztencia Congress Bureau, 2009, pp. 200-203. ISBN 978-963-06-7716-5.
- [11] KOLENOVSKY, Filip. *Komunikacni system pro varovani osob pred nebezpecim s využitím IP telefonie*. Ostrava, 2012. Diploma Thesis. VSB-TEchnical University of Ostrava.
- [12] VOZNAK, M. and J. ROZHON. SIP Registration Stress Test. In: *6th International Conference on Communications and Information Technology (CIT '12)*. Athens: WSEAS, 2012, pp. 95-100. ISBN 978-1-61804-077-0.
- [13] Another Strike Against WEP Security. *SpamLaws* [online]. 2009. Available at: <http://www.spamlaws.com/wep-security.html>.

About Authors

Filip REZAC was born in 1985 and his M.S. received from VSB–Technical University of Ostrava in 2009. Since 2009, he has continued in Ph.D. study in department of Telecommunications and he gained position of assistant professor with the same department in 2010. His research interests are focused on Voice over IP technology, Network Security and Speech Quality. Since 2008, he has become a researcher with Department of Multimedia in CESNET.

Filip KOLENOVSKY was born in 1988. In 2010, received a Bachelor title in VSB–Technical University of

Ostrava, Faculty of Electronics and Computer Science, Department of Informatics. Two years later he received the M.Sc. title focused on computer science in the same workplace. His professional interests include programming of web applications and programming for the Linux operating system.

Martin MIKULEC received the M.Sc. degree from VSB–Technical University of Ostrava, Faculty of Electrical Engineering and Computer Science in 2011. Currently, he is working toward the Ph.D. degree at the Department of Telecommunication. His research is focused on advanced services in IP telephony.

Miroslav VOZNAK is an associate professor with Department of Telecommunications, VSB–Technical University of Ostrava, Czech Republic. He received his M.Sc. and Ph.D. degrees in telecommunications, dissertation thesis “Voice traffic optimization with regard to speech quality in network with VoIP technology” from the Technical University of Ostrava, in 1995 and 2002, respectively. Topics of his research interests are Next Generation Networks, IP telephony, speech quality and network security. He was involved in several FP EU projects. At present, he is also working for the Czech National Centre of Excellence IT4I (IT for Innovations).

Jan ROZHON was born in 1986. In 2008, received a Bachelor title in VSB–Technical University of Ostrava, Faculty of Electronics and Computer Science, Department of Informatics. Two years later he received the M.Sc. title focused on telecommunications in the same workplace. Currently in the doctoral study he focuses on Voice over IP technology, namely performance of SIP network elements.

Jiri VYCHODIL was born in 1984 in Olomouc. In 2007 finished bachelor study in VSB–Technical University of Ostrava, branch Telecommunication Technologies. Engineer study finished in 2009, branch Mobile Technologies. Currently works as doctoral student in the Department of Telecommunication at the University.

Karel TOMALA was born in 1984. In 2007, received a Bachelor title in VSB–Technical University of Ostrava, Faculty of Electronics and Computer Science, Department of Telecommunications. Two years later he received the M.Sc. title focused on Telecommunications in the same workplace. Currently in the doctoral study he focuses on Voice over IP technology and Call Quality in VoIP.

Jakub SAFARIK received his M.Sc. degree in telecommunications from VSB–Technical University of Ostrava, Czech Republic, in 2011 and he continues in studying Ph.D. degree at the same university. His research is focused on IP telephony, computer networks and network security. He is with CESNET as a researcher since 2011.